Homework 4

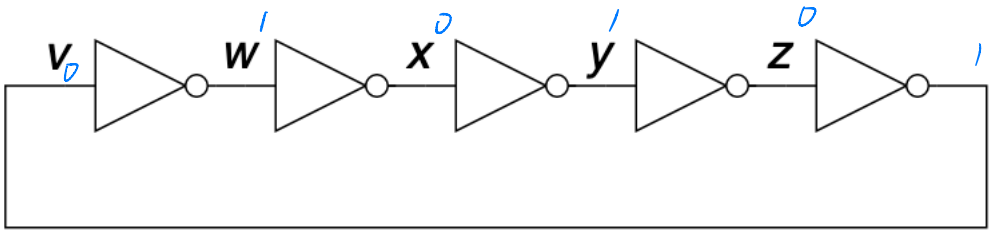
EECS 270 Winter 2020

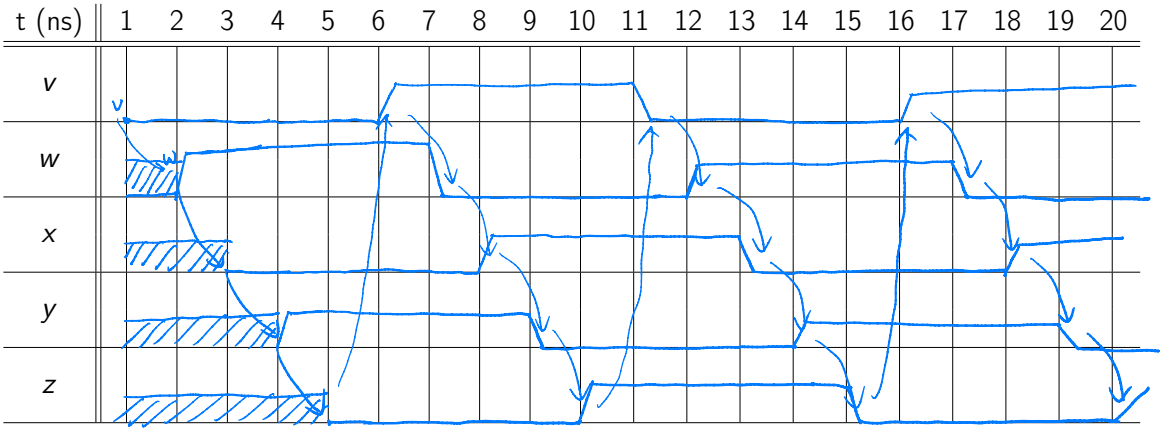
Due Friday, February 14 @ **11:59 PM** on Gradescope

This is an individual assignment, all of the work should be your own.
Write neatly or type and show all your work for full credit.
Have your name and unique name on the front page of your submission.

1. **[20 points]** Implement $F(x, y, z) = x'y' + xz' + x'yz + y'z$ as a digital logic circuit. Your available parts are the following: 2 inverters, a 2-4 decoder, and a 2-1 mux. *Hint: Think about how to split the function into two parts that can be wired into the 2-1 mux. As in, is there a way to expand the function into two cofactors: one for a variable and another for the complement of that variable?*

2. **[12 points]** The following circuit is not a combinational logic circuit, as its output affects its input. Assume that the delay of a NOT gate is 1 ns.





   a. **[8]** Draw the timing diagram with causality arrows for this circuit until t=20 ns. Assume that v, w, x, y, and z's initial values are unknown, and at t=1 ns v transitions to 0. While a value of a signal is still unknown, shade in the region of the diagram for that particular signal.

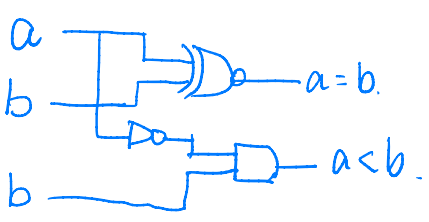
   b. **[2]** What is the period of v? 10 ns

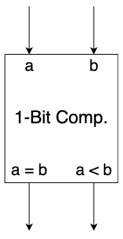
   c. **[2]** What is the frequency of v? $\frac{1}{10\text{ns}} = \frac{1}{10\times10^{-9}\text{s}} = 10^8\text{Hz} = 100\text{MHz}$

3. **[18 points]** Comparators:

   a. **[6]** Design the logic gate equivalent for the 1-bit comparator as shown below. The outputs are 1 when they are true: if $a = b$, then that output will be 1, whereas if $a < b$, that output will be 1.

$a=b \rightarrow a \odot b$
$a<b \rightarrow a'b$.





a	b	a<b.
0	0	0
0	1	1
1	0	0
1	1	0

$a=b$ 1 $a \odot B$. $a \odot B$.

$a'b$.

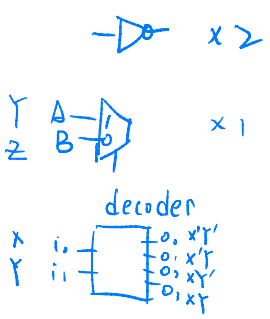
Homework 4

EECS 270 Winter 2020

Due Friday, February 14 @ **11:59 PM** on Gradescope

This is an individual assignment, all of the work should be your own.
Write neatly or type and show all your work for full credit.
Have your name and unique name on the front page of your submission.

1. **[20 points]** Implement $F(x, y, z) = x'y' + xz' + x'yz + y'z$ as a digital logic circuit. Your available parts are the following: 2 inverters, a 2-4 decoder, and a 2-1 mux. *Hint: Think about how to split the function into two parts that can be wired into the 2-1 mux. As in, is there a way to expand the function into two cofactors: one for a variable and another for the complement of that variable?*



$x'y'(z+z') + x(y+y')z' + x'yz + (x'+x)y'z$

$= x'y'z + x'y'z' + xyz' + xy'z' + x'yz + x'y'z + xy'z$

$= 001 \quad 000 \quad 110 \quad 100 \quad 011 \quad 001 \quad 101$

$m_1 \quad m_0 \quad m_6 \quad m_4 \quad m_3 \quad m_1 \quad m_5$

$x'F(0,y,z) + xF(1,y,z)$

$(yz)' \quad y'+z'$

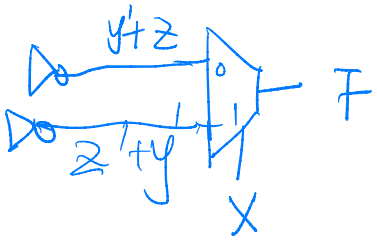
$(yz')' \quad y'+z$

$x'(y'+yz+y'z) + x(z'+y'z)$

$z'+y'$

$y'+yz$

$x'(y'+z) + x(y'+z)$



$x'y' + x'z + xy' + xz$

$x'A + xB$

b. **[12]** Design and draw a digital logic circuit that takes in 2 3-bit unsigned inputs, and outputs the lesser of the two values (the circuit can output either value when they are equivalent). You may use instances of the 1-bit comparator part above (as a box, not the internal gates you drew for part (a)), AND, OR, and NOT gates, along with a 3-bit 2-1 mux. The inputs to the circuit are 3-bit values A[2:0] and B[2:0], and the output C[2:0].

4. **[10 points]** Design the following functions using only tri-state gates and their variants–*i.e.,* tri-state buffer, tri-state active-low buffer, tri-state inverter, tri-state active-low inverter.

   a. **[5]** 2-input Mux with inputs I_1, I_0, selector S, and output O.

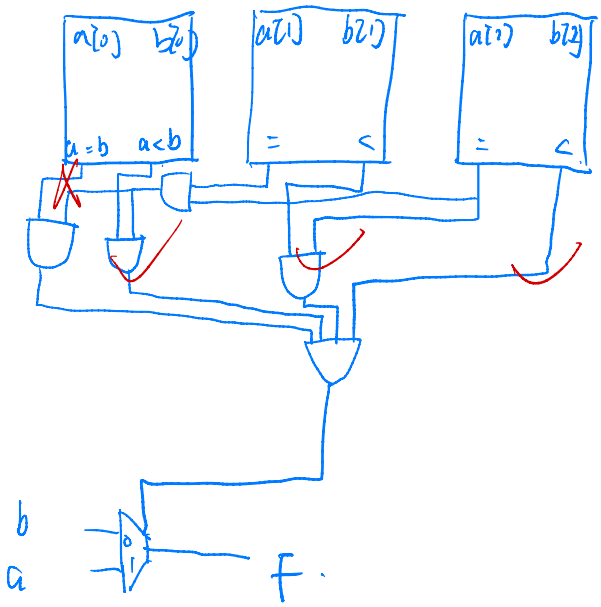
   b. **[5]** 2-input AND with inputs a and b, and output F.

5. **[12 points]** Convert the following decimal (base-10) numbers into (a) magnitude in binary, (b) signed magnitude, (c) 1s' complement, and (d) 2's complement: (you do NOT have to show how you converted to binary magnitude, it's more of a step to help you get to the other forms)

   a. **[4]** 18

   b. **[4]** -5

   c. **[4]** -107



6. **[10 points]** Design a 3x8 decoder with enable signal with only the following gates available to you: AND, OR, NOT.

7. **[18 points]** Design and draw a digital logic circuit that analyzes a 6-bit unsigned input $X[5:0]$. RESTRICTION: Your circuits may use however many 2-input XOR and XNOR gates you need to implement it.

    a. **[4]** If X is even, set the output $P1$ to 0 and if it's odd, set $P1$ to 1.

    b. **[10]** Determine if the number of 1's in the number is even or odd. If it is even, set $P2$ to 0. If it is odd, set $P2$ to 1.

    c. **[4]** Use the outputs of $P1$ and $P2$ to set a final output, P, which satisfies the following conditions:

        i. If X is an even number, P will be 0 if there are an even number of bits in X that are 1, and 1 if there are an odd number of bits that are 1.

        ii. If X is an odd number, P will be 0 if there are an odd number of bits in X that are 1, and 1 if there are an even number of bits that are 1.